ARTÍCULOS DE TECNOLOGÍA

Como crear una buena contraseña

yan-orestes
09/12/2021

Como usuario de las múltiples aplicaciones web, con seguridad ya enfrentaste algún problema con una contraseña. Tal vez la hayas olvidado, o tal vez hayas tenido una cuenta invalida.

Hace unos días, decidí crear una cuenta en una red social que a mis amigos les gusta. Cuando escribí la contraseña que quería, un mensaje de error apareció en la pantalla diciendo que era muy débil. Por lo visto la contraseña de todos modos no iría a funcionar.

Recordé aquella regla que todos escuchamos en algún momento, que una buena contraseña debe tener caracteres en minúscula, mayúscula, números, símbolos y rehice ella. Esta vez, ¡funciono!

¡Todo parecía bien! Pero dos semanas después, recibo una notificación, alguien había invadido mi cuenta. Obviamente, ¡quedé indignado! Mi contraseña seguía aquellas reglas que yo aprendí. La contraseña era `Cl@ve123`...

Después de investigar, descubrí que mi cuenta pudo haber sido invadida a través de un **ataque de diccionario**, el _dictionary attack_. Con esa técnica, el atacante, a través de un proceso automático, intenta ganar acceso con múltiples contraseñas diferentes. Esas contraseñas generalmente están en un archivo llamado de **diccionario de contraseñas**, existen varios de esos disponibles en la Internet.

¿Pero cuál es el problema de esas contraseñas? ¿Cómo entonces, podemos crear una buena contraseña? ¿Cuáles son los aspectos importantes que debemos tener en cuenta?

Mayúscula, minúscula, número, puntuación ¿es eso?

Mi contraseña `Cl@ve123` contiene tres letras minúsculas, una mayúscula, un símbolo y tres dígitos numéricos. Recordando lo que aprendimos en nuestras clases de matemática, podemos calcular cuantas posibilidades de contraseña existen.

Tenemos **26** posibilidades de letras minúscula, más **26** de mayúsculas, más **10** caracteres numéricos, más por vuelta de **28** caracteres de puntuación, totalizando **90** caracteres posibles.

Como nuestra contraseña tiene 8 caracteres de largo, la cuenta queda $\mathbf{90^8}$, ¡que es igual a **4.304.672.100.000.000** posibilidades!

Digamos que el atacante consigue probar 1000 contraseña por segundo. Para probar todas esas posibilidades, el necesitaría de **más de 136 mil anos**.

> *"Wow! Un número gigantesco! Pero si es así, ¿cómo consiguieron descubrir la contraseña? ¿Fue suerte del atacante?"*

La verdad, lo que pasa es que toda esa cuenta no se aplica a nuestra contraseña, ¡**porque ella no es aleatoria**! Tenemos más de 4 cuatrillones de posibilidades en total, pero la nuestra no es apenas más una en ese montón. Esa contraseña es deducible, y es así que el ataque de diccionario trabaja.

Siguiendo las reglas de una manera efectiva

El ataque de diccionario, diferente de otros tipos de ataque, como el **ataque de fuerza bruta** (*brute force*), no tiene el objetivo de intentar todas las posibilidades que existen dentro de un contexto. Lo que él hace , dadas algunas palabras o grupos de palabras comunes en contraseña, intentar el acceso con varias posibilidades para cada palabra, considerando substituciones y modificaciones comunes.

O sea, el ataque no va sólo verificar si `CL@ve123` funciona, sino también `cl@ve123`, `cl@VE123` y, eventualmente, `Cl@ve123`. **Eso no significa que caracteres especiales y de grupos diversos no hacen diferencia, pero es una ilusion entenderlos como garantia de seguridad**.

Eso quiere decir que si nuestra contraseña tiene una base tan simple y usual como `Clave123`, substituciones comunes como de a para @ y cambios de género de hecho no aumentaran mucho el nivel de seguridad de nuestro secreto. Entonces, ¿ qué debemos cambiar?

Si juntamos un poquito de todo, podemos seleccionar una palabra aleatoria, como **cordero**, e intentar hacer algunas modificaciones no tan usuales. Con algunas alteraciones, llegue en `C0rd3ro%6` una contraseña ya bien más segura comparada a la anterior.

Espere un tiempo, luego, finalmente logué en mi cuenta. Coloque mi email y... ¿cuál era la contraseña, al final? *Cordero* alguna cosa, uno de los **o** se volvía **0** y también había algún símbolo al final, yo creo... ¡Que complicado!

¿Y ahora? ¿Será que la complejidad de esa contraseña es compensada por la seguridad que ella trae? ¿No habría un método mejor para nosotros los usuarios?

contraseñas complejas Vs contraseñas largas

Todo ese problema de memorización y utilización que ese tipo de contraseña más compleja trae, es la base de las principales problemas. Randall Munroe, caricaturista del *webcomics*

xkcd, creo una caricatura en 2011 jugando un poco con el tema, que sigue siendo muy relevante.

Con un aire cómico, la caricatura nos muestra, con cálculos de entropia, que contraseñas como `C0rd3ro%6` no son tan efectivas en el requisito de seguridad, a pesar de ser difíciles de memorizar. En contrapartida, contraseñas más largas, con varias palabras aleatorias, tienden a una mayor facilidad de memorización, al mismo tiempo que un nivel mayor de seguridad.

Frases-contraseñas con palabras retiradas aleatoria mente de un diccionario pueden ser creadas con auxilio de herramientas como generadores de palabras aleatorias y memorizadas a través de diferentes técnicas (que no deben ser muy difíciles), nos asegura una contraseña poco convencional y, mismo así, ¡bastante efectiva!

Pero ¿ entonces? ¿Creamos nuestra frase-contraseña larga, con diversas palabras aleatorias, memorizamos y es eso?¿ Podemos usarla para todas nuestras cuentas entonces?

contraseñas únicas para cada cuenta

Infelizmente, los ataques de diccionario no son el único peligro que nuestras contraseñas enfrentan, hoy en día. Hay casos en que la seguridad ni depende de nosotros los usuarios, como cuando alguna plataforma almacena las contraseñas en texto plano, en vez de almacenar una hash generada a partir de esa contraseña.

Hay aún casos que, a pesar de depender de nosotros, no dependen exactamente de la complejidad de nuestra contraseña, como con *malwares que capturan las teclas digitadas*, o hasta mismo un caso bobo de alguien observando el teclado en cuanto digitamos.

Por cuenta de eso, no podemos confiar en una única contraseña para todo. Lo ideal sería que tengamos contraseñas completamente diferentes (esto es, que no sigan un mismo patrón) para cada servicio en que creamos una cuenta. Así, en el caso que una de nuestras contraseñas fuera descubierta, por cualquier motivo que sea, nuestras otras cuentas no estarían comprometidas!

Considerando todo lo que ya aprendimos, podemos decidir por crear frases-contraseña diferentes, con palabras diferentes, para cada cuenta que creamos. Entretanto, eso generalmente no es viable, al final ¿quién va querer estar memorizando decenas de frases aleatorias?

Por cuenta de eso, muchos usuarios acaban optando por la simplicidad de una única contraseña debil para todos los servicios, en vez del método más seguro. Pero ¿será que no tenemos como encajar utilidad y seguridad en ese caso?

Usando administradores de contraseña para evitar la memorización

Después de tanto buscar por la mejor opción de contraseña, nuevamente caemos en el problema de la memorización... Para nosotros, lo mejor sería si realmente no necesitamos memorizar nada, o por lo menos no muchas cosas, pero aún así tuviésemos contraseñas seguras.

Podríamos guardar todas nuestras contraseñas en un único archivo, para apenas copiar y colocar ellas cuando fuésemos a usar. Pero dejar en un simple archivo suelto sería un gran riesgo. ¿Y si hubiera un método mejor, qué garantía la seguridad de esas contraseñas almacenadas?

Para eso, existen los **administradores de contraseña**, como el KeePass y el LastPass, que nos ayudan, de forma segura y organizada, a almacenar todas nuestras contraseñas.

Podemos configurar el acceso a ellas con contraseñas (que pueden seguir las recomendaciones que trabajamos a lo largo del post!) y hasta con archivos claves, en el caso del KeePass, necesarios para la visualización del banco de datos. Pero si ahora vamos a guardar nuestras contraseñas en un administrador, ¿cómo debemos crearlas al fin?

Usando contraseñas aleatorias como garantía

Ahora que no necesitamos más memorizar contraseñas sin necesidad, podemos abusar aún más del aspecto de seguridad, generando nuestra contraseña a través de algoritmos de aleatoriedad, lo que, como calculamos más arriba, aumenta intensamente el nivel de seguridad de nuestras contraseñas.

Con un buen largo y una buena variedad de caracteres (**que no necesitan restringirse a los ASCII, si la plataforma hace un buen trabajo de *hashing* de contraseñas**), podemos crear contraseñas (**casi) imposibles de descifrar con métodos de diccionário/fuerza bruta.

Una contraseña generada por ese generador aleatorios de contraseñas en emoji, por ejemplo, podría demorar más de **1,49 X 10^{52} veces la edad de la Tierra** para ser descifrada, si todas las personas del planeta intentasen simultáneamente descifrarla con un

computador diferente a un millón de tentativas por segundo. Básicamente, es improbable que una contraseña de esas sea descifrada.

```
<iframe src="https://yanorestes.github.io/emoji-pwd/" name="myiFrame" scrollin
```

Para los casos en que necesitamos de un acceso más dinámico a alguna cuenta, que no se restrinja a un computador, podemos usar de los otros consejos que aprendimos en el post.

Además de eso, es importante, **siempre que es posible**, utilizar la herramienta de **autenticación de múltiplos factores**, o _multi-factor authentication_. Con esa configuración, el acceso a una cuenta no depende apenas de la contraseña, más también de factores externos diversos.

Una posibilidad es, por ejemplo, exigir el código de login por SMS. Así, estaremos aún más seguros, mismo que nuestra contraseña sea comprometida.

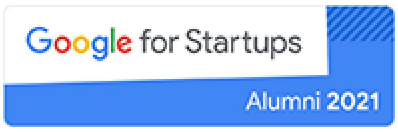
Mayor seguridad de datos para mayor seguridad nuestra

A pesar de muchas veces pensemos que no hace mucha diferencia utilizar una contraseña (que sabemos que es) muy débil, eso no es verdad. No tomar una actitud en pro de la seguridad de nuestros datos puede acabar en grandes perjuicios en nuestra vida.

Tratándose de las contraseñas, entendemos lo que vuelve una contraseña segura (es lo que puede volver insegura). También aprendemos técnicas efectivas de creación de contraseña que nos pueden auxiliar bastante como usuarios de múltiples servicios y plataformas.

ALIADOS

En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

CURSOS

Cursos de Programación
Lógica de Programación | Java

Cursos de Front End
HTML y CSS | JavaScript | React

Cursos de Data Science
Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

Cursos de DevOps
Docker | Linux

Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics | Liderazgo y Gestión de Equipos | Startups y Emprendimiento